EE318 Electronic Design Lab I Project Report, EE Dept., IIT Bombay, April 2006

SMS enabled PSTN phone

Group No. B8

Jagat Narayan Sabat (03007027) <jagat@ee.iitb.ac.in>
Rahul Singh Thakur (03007026) <rahulsingh@ee.iitb.ac.in>
Sandeep Anand (03007002) <sandeepa@ee.iitb.ac.in>

Supervisor: Abhay Karandikar
Course Instructors: P.C.Pandey and L.R.Subramanium

Abstract

The aim of the project is to provide Short Messaging Service (SMS) facility to landline phones connected through PSTN (Public Switched Telephone Network).lines. This is a two sided problem – first the clients (normal telephones) should have a module for enabling writing and sending SMSs. Second, there should be a server side to process the SMSs and send them to their appropriate destination.

1. Introduction

SMS is one of the most popularly used service among mobile phone users because of the fact that message can be viewed by the user at his/her convenience. In our project, we will try to provide this facility to landline (PSTN) users.

Specifically, our task is to build an SMS center (server). An SMS that is sent is transmitted to an SMS center and from there, it is forwarded to the receiving mobile phone or a landline phone, or stored, if the phone is not available. It then waits for its availability and then sends the message as soon as the destination phone becomes free.

2. Basic Approach

We have designed a hardware interface between transmission line and the SMS center (computer) which converts an encoded incoming message into 8-bit data packets and send it to computer and vice versa in the case of data transmission from the server. The incoming message consists of a Dual Tone Multi Frequency (DTMF) encoded destination telephone number followed by the actual message encoded in Frequency Shift Keying (FSK) format. After decoding, the hardware will send the data packets serially to computer using RS232. The computer will process that data and store it. Then it will try to connect to destination phone and, if connected, will send the message.

The software part includes various codes written for the microcontrollers and the code for storing, processing and transmitting back to the circuit.

3. Design of circuit

The telephone signaling chip that we are using can only be used for synchronous serial data transmission and reception while the computer can only do asynchronous transmission and reception because of the presence of UART in computer. So to overcome this problem, we are using 2 microcontrollers, one is used to establish a synchronous serial communication with the telephone signaling chip; and the other microcontroller for establishing asynchronous serial communication with the computer. The two microcontrollers in turn are connected in parallel and are controlled using some handshake signals between them. So we needed to use microcontrollers with sufficient number of ports and hence chose AT89C52 microcontroller. To establish the communication between computer and microcontroller, we have used RS232 with MAX232 IC for converting RS232 signals to standard TTL logic.

A basic circuit block diagram is as given below.

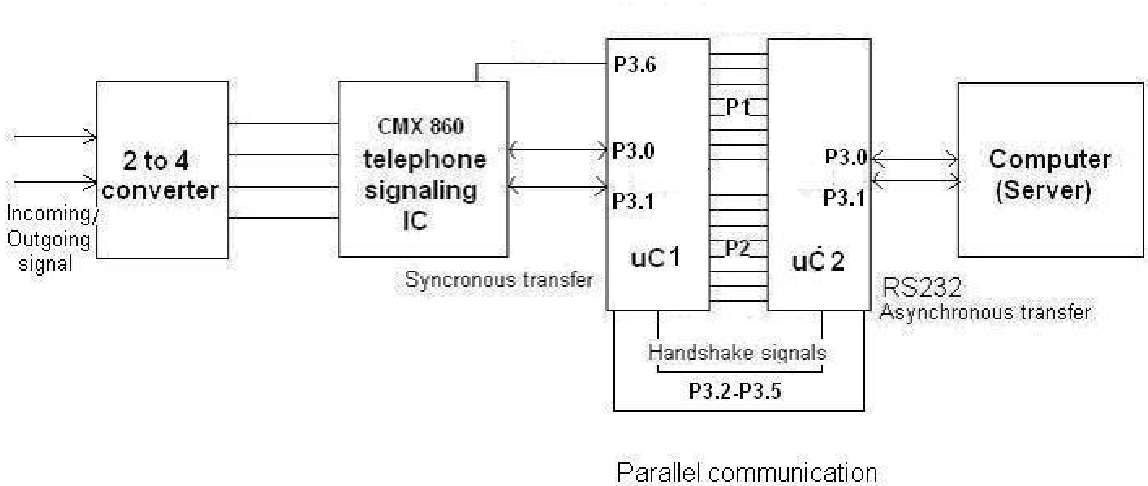

Fig 1 : Basic Block Diagram

CMX860

We need a telephone signaling chip to decode the DTMF encoded destination telephone no. and the actual FSK encoded message and encode the same while sending it to a client. For this purpose, CMX 860 telephone signaling chip is being used. This 28 pin IC consists of a DTMF encoder and decoder, FSK modulator (1200bps or 75bps) and demodulator plus call progression circuitry with analogue switching between line and phone interfaces. Ring detection, local phone off-hook detection and a relay driver for line hook-switch operation are also provided under the control of 'C-BUS'. It can also be configured using serial data from microcontroller using C-BUS.

The CMX860 transmit and receive operating modes are independently programmable. In FSK mode it can communicate both in synchronous and asynchronous mode. For data transmission we are using the IC in asynchronous mode with 1200bps,1 stop bit and no parity bit. Since in our case, we are making the server side and don’t have a hook, therefore we are not using a hook detector.

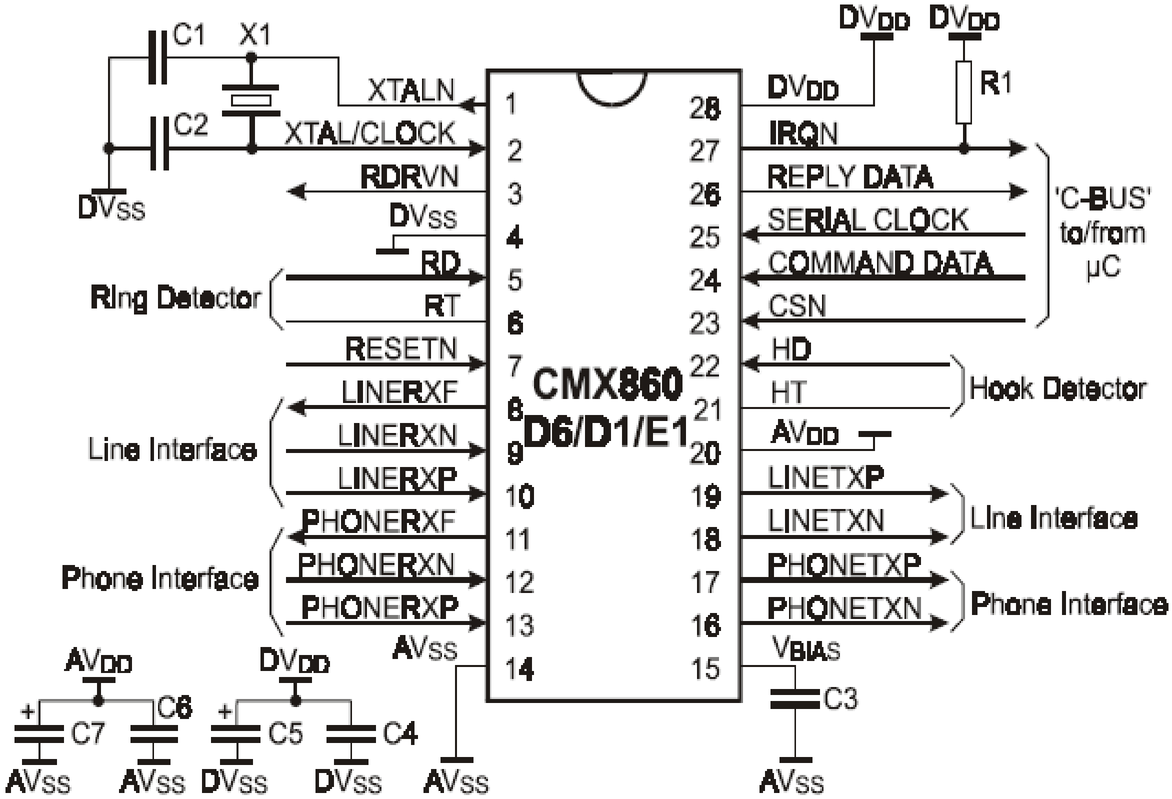

Fig 2 :CMX 860 Pin diagram [2]

The Ring Detector

We are using a ring detector as shown in Fig.3. The ring signal is usually applied at the subscriber's exchange as an ac voltage inserted in series with one of the telephone wires and will pass through either C20 and R20 or C21 and R21 to appear at the top end of R22 (point X in Figure 3) in a rectified and attenuated form.
The values of some of the components is as given : R20, 21, 22 : 470kΩ, C20 : 21 0.1μF, C22 : 0.33μF, R24 : 470kΩ
Setting R23 to 68kΩ will guarantee detection of ringing signals of 40Vrms and above for DV_{DD} over the range 3V to 5V.

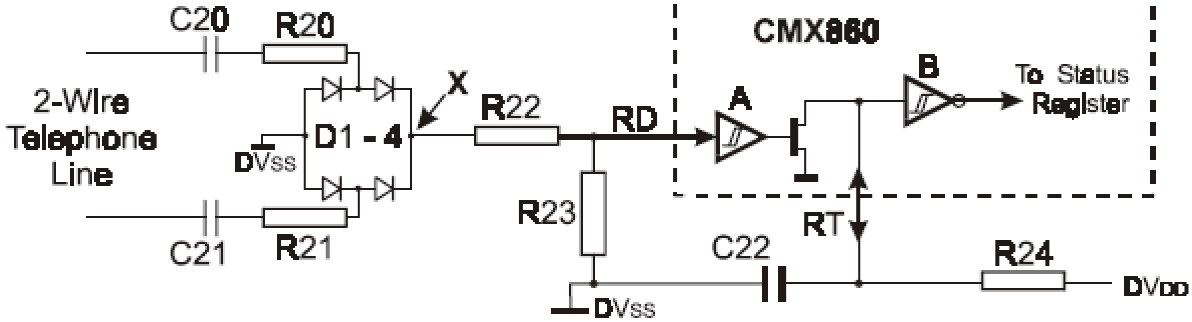

Fig 3 : Ring Detector circuit [2]

AT89C52 microcontroller

An 89C52 microcontroller is a CMOS 8-bit microcomputer with 8K bytes of Flash programmable and erasable read only memory (PEROM), 256 bytes of RAM, 32 I/O lines, three 16-bit timer/counters, a six-vector two-level interrupt architecture,a full-duplex serial port, on-chip oscillator, and clock circuitry. The pin diagram of an AT89C52 IC is as shown in Fig.4.

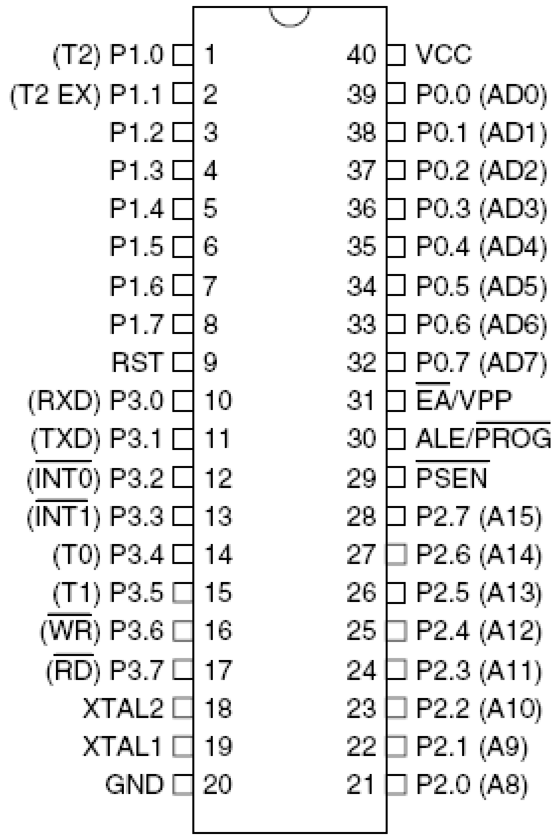

Fig 4 : AT89C52 Microcontroller [1]

Two microcontrollers are used – one is connected to the computer using the RS232 cable with the MAX232 to interconnect the RS232 to the microcontroller. This microcontroller does asynchronous data transfer with the computer. The second microcontroller is connected directly with the CMX860 IC and does synchronous serial data transfer. The two microcontrollers are connected back to back with port 2 of each joined for parallel data transfer.

The handshake signals used for the parallel data transfer are :

Pin	uC1	uC2
P3.2	out (RTS1)	in (RTS1)
P3.3	out (DR1)	in (DR1)
P3.4	in (RTS2)	out (RTS2)
P3.5	in (DR2)	out (DR2)

RTS : Ready To Send

DR : Data Received

MAX 232 and RS 232

In microcontrollers, we require either TTL or CMOS logic levels, so we need the RS-232 levels back into 0 and 5 Volts. The MAX 232 is used to interconnect the RS232 cable to a microcontroller. It is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. It also converts the +10 V and -10 V logic of RS232 to 0 to 5 V TTL logic levels.
The pin diagram of MAX232 is given in Fig 5.

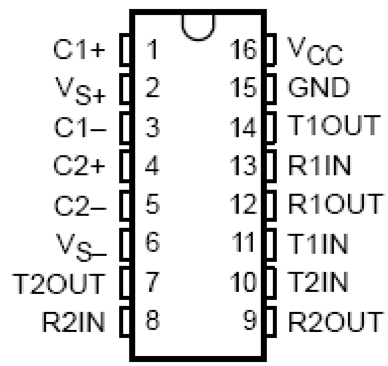

Fig 5: MAX 232 IC Pin diagram [3]

The working of the MAX232, RS232 and the connection of the microcontroller to the computer is as shown below.

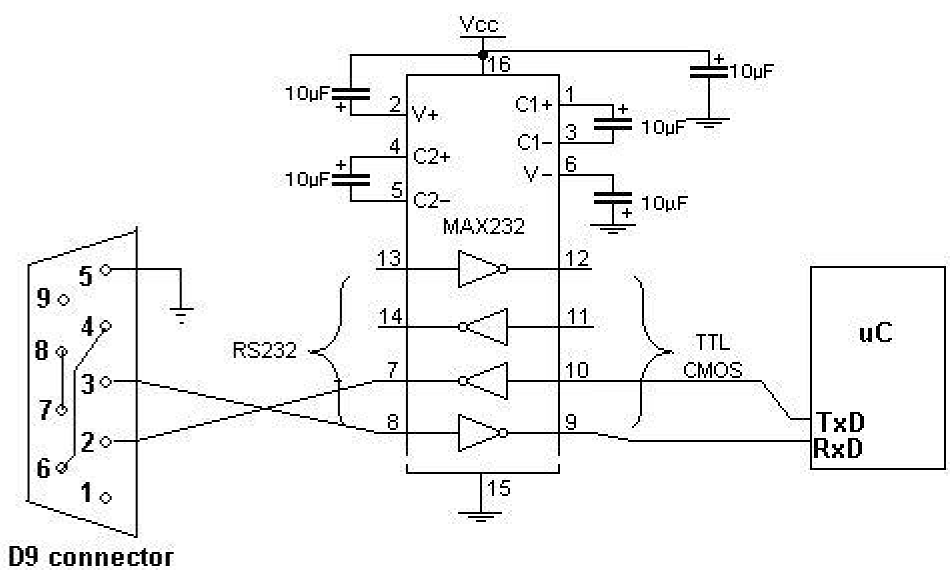

Fig 6 : Computer to uC interface

4. Test Procedure

The whole test procedure was done in various parts.

Asynchronous serial communication between computer and microcontroller

An assembly language code was burned into a microcontroller and a Turbo C++ program to emulate a hyper-terminal was written on the computer. The circuit was set up as shown in figure 6. We get that whatever was sent from the microcontroller as a hexadecimal number was received by the C++ program as a decimal number and a decimal number sent from the computer was received in its hexadecimal form. The baud rate used was 9600 bps.

Synchronous serial communication between CMX860 and microcontroller

The CMX860 was initialized by sending data to set it up for sending and receiving and then data was sent from the microcontroller. We get that whatever was sent was received by the telephone signaling IC.

We next integrated the two independently functioning parts by connecting the two microcontrollers by a parallel port connection and other pins as handshake signals. The codes for initializing the CMX860 was then sent from the computer and we got Vbias =2.2V ,then we configured it to send continuous 1`s and got sinusoidal output of 1300Hz at the line output pin.

Ring detector circuit

The circuit for the ring detector is as shown in Fig.3. It was tested with a small telephone exchange which sends 90V 20Hz signal as a ring. The ring was detected and we confirmed that by reading the status register (E6h) of CMX860.

5. Current Status

Our circuit integrated on the breadboard is able to send and receive data from client. 2to4 is not working up to the mark so we are connecting the server and the client using 4 wires.

6. References

[1] “AT89C52 datasheet, 8-bit Microcontroller with 8K Bytes Flash”, ATMEL, http://www.atmel.com/dyn/products/datasheets.asp?family_id=604

[2] “CMX860 datasheet, Telephone Signaling Transceiver”, CML Microcircuits, http://www.cmlmicro.com/

[3] “MAX232 datasheet, Dual EIA-232 Driver/Receiver”, MAXIM, http://pdfserv.maxim-ic.com/en/ds/MAX220-MAX249.pdf

[4] “RS232 serial communication”, http://beyondlogic.org/serial/serial.htm